RealAudio Content Creation Guide

RealAudio Encoder
Release 2.0

Progressive Networks, Inc.

Information in this document is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Progressive Networks, Inc.

© 1995, 1996 Progressive Networks, Inc. All rights reserved.

Apple and Macintosh are trademarks of Apple Computers, Inc.

Cool Edit is a trademark of Syntrillium Software Corporation

Netscape Netsite is a trademark of Netscape Communications Corp.

O'Reilly Website NT is a trademark of O'Reilly Associates, Inc.

RealAudio, the RealAudio logo, and Progressive Networks are trademarks of Progressive Networks, Inc.

UNIX is a registered trademark of X/Open Consortium

Windows and Windows NT are registered trademarks of Microsoft Corp.

Printed in the United States of America.

Part #: S200-008C-960318.

Table of Contents

CHAPTER 1: PRODUCING REALAUDIO CONTENT 1

Overview 2
Introduction 2
Recording Audio Files 3
Creating RealAudio Files 5
The RealAudio Encoder 5
Menu Bar 6
Source and Destination Panels 7
Progress Information 8
Description Panel 8
Options Panel 9
Starting the Encoding Process 10
Drag and Drop Encoding 10
Command-line Encoding 10
Cutting and Pasting RealAudio Files 12
Bandwidth Negotiation 13

CHAPTER 2: WEB SITE CONFIGURATION 15

RealAudio Metafiles 15
Creating Metafiles 16
Customizing Links to Audio Content 17
RealAudio Plug-ins 19
System Requirements 20
Using the Plug-in: The Basics 20
The SRC Attribute 20
The WIDTH and HEIGHT Attributes 21
Using the <NOEMBED> Tag 21
Custom RealAudio EMBED Tag Attributes 22
The CONTROLS Attribute 22
The NOLABELS Attribute 24
The AUTOSTART Attribute 25
Making Plug-ins Work Together: the CONSOLE Attribute 25

Using Plug-ins with HREF Tags 26
Making the Most of Your Content 27

CHAPTER 3: PRODUCING LIVE EVENTS 29
Server Setup for Live Events 29
Live Events with RealAudio 14.4 and 28.8 32
Prerecorded "Live" Delivery 32

CHAPTER 4: SYNCHRONIZED MULTIMEDIA 35
Creating Synchronized Multimedia Events with RealAudio 35
Creating the Input File 36
Generating an Event File Using the CEVENTS Command-Line Tool 37
Placing Event Files on Your Server 37
Using Synchronized Multimedia from Local Files 37
Frames: The Basics 37
Frame Document 38
Synchronizing frames and audio 38
More Information on frames 39

APPENDIX A: MAN PAGES 41
lta 41
Name 41
Synopsis 41
Description 41
slta 43
Name 43
Synopsis 43
Description 43
cevents 44
Name 44
Synopsis 44
Description 44
File Format 44

INDEX 45

Chapter 1: Producing RealAudio Content

Welcome to the world of RealAudio, the premier Internet audio delivery system. By setting up a RealAudio Server and providing audio libraries, you are becoming an active participant in the exciting new world of real-time audio on the World Wide Web.

RealAudio technology provides real-time audio-on-demand and consists of three main components: the RealAudio Player, the RealAudio Server, and the RealAudio Encoder. The RealAudio Encoder enables the generation of RealAudio content from raw audio input as well as from pre-recorded audio files in a variety of formats.

The RealAudio Player lets you listen to audio files that have been encoded into the RealAudio format. The Player plays RealAudio files in real-time across a TCP/IP network. It is available for download from the Progressive Networks Website at `http://www.realaudio.com`. RealAudio Players are available for Microsoft® Windows®, Apple® Macintosh® and UNIX operating systems.

The RealAudio Server enables audio streams to be sent over the Internet to RealAudio Players. Server operation is similar to that of Internet Web servers. RealAudio Servers are available for Microsoft® Windows NT®, Apple® Macintosh® and UNIX platforms.

The PC version of the RealAudio Encoder enables users and Server operators to encode standard PC and UNIX audio files into the RealAudio format. RealAudio files containing compressed audio data can then be streamed by the Server in real-time over modem connections.

This documentation provides information on the operation of your RealAudio Encoder, the configuration of Web sites, and the creation of multimedia RealAudio content.

Overview

The RealAudio System enables the creation, presentation and transmission of real-time on-demand audio across the Internet. Three separate processes are necessary to take full advantage of the RealAudio System:

1. Encoding the audio content using the RealAudio Encoder.
2. Providing access to the content using various methods of presentation, such as:
    - Using links to `.ram` files
    - Using RealAudio Plug-ins
    - Using RealAudio Synchronized Multimedia
    - Doing live broadcasts
3. Transmitting the RealAudio content.

This manual is intended to explain the details of encoding and presenting RealAudio content. Please see the *RealAudio Server Administrator's Guide* for extensive information on RealAudio transmission.

Introduction

Current Internet data transmission rates necessitate signal compression prior to transmission of real-time audio. The RealAudio Encoder performs this function within the RealAudio System; the RealAudio Player can only play files that have been encoded using the RealAudio Encoder, and the RealAudio Server can only serve files that have been encoded using the RealAudio Encoder. Thus, the RealAudio Encoder is the bridge between the RealAudio Server and the RealAudio Player.

RealAudio content may be created either from previously recorded digital audio files, or from an external audio source. Pre-recorded files may be in .au, .wav and raw .pcm audio formats. External audio may be fed from a recorded source, such as DAT (Digital Audio Tape), CD (Compact Disc), or directly from a mixing console or microphone. Prior to encoding, pre-recorded audio should be preprocessed using one of the various sound

editing utilities available to maximize the audio quality of the encoded content. Similarly, when encoding live source audio, the input levels should be prepared and tested as before any professional recording session.

The RealAudio Encoder can encode using two different algorithms. The 14.4 algorithm provides optimal sound quality to end-users with a 14.4 Kbps modem connection to the Internet. This algorithm has a high compression ratio optimized for speech transmission. RealAudio 2.0 now includes a 28.8 algorithm, which provides RealAudio to end users with a 28.8 Kbps or faster connection to the Internet. This format, although still highly compressed, takes advantage of the added bandwidth and delivers monophonic FM quality audio.

Providing Web site access to your RealAudio content is similar to creating links between Web pages. When accessing RealAudio content, however, your links are to RealAudio metafiles rather than to other Web pages. Although RealAudio files can also be first downloaded and then played, providing RealAudio content in real-time requires a RealAudio Server. Metafiles tell a visitor's Web browser to launch either the RealAudio Player or the RealAudio Plug-in and to play the desired file. Chapter 2, *Web Site Configuration*, provides information on RealAudio metafiles and on creating access to your RealAudio content.

The RealAudio System can broadcast "live" to the Internet. It can also be used as part of a multimedia presentation. "Live" broadcasting means that visitors to your Web site can listen to your audio from the current point in the broadcast, just like listening to a radio broadcast. A multimedia event is a combination of your RealAudio content and your Web page content. Both of these capabilities are provided with the Windows Live Encoder, which is distributed with the RealAudio Server. To purchase a RealAudio Server please contact Progressive Networks' sales department at 1-800-230-5975 or

http://www.realaudio.com/products/server.html

Recording Audio Files

The RealAudio Encoder compresses the audio data stream and prepares it for use with the RealAudio Server. This compression is "lossy", which means that some of the information contained in the original audio data stream is removed. To ensure that you will get optimal audio quality after encoding, you must start with a high fidelity recording which utilizes the full dynamic range and has a high signal to noise ratio. Using Digital Audio Tapes

(DATs) or Compact Discs (CDs)[1] as your source, for example, will result in high quality RealAudio files. In addition, you will have better results if you use simpler audio signals rather than very complex signals.

While it is possible to record directly from a source, pre-processing the sound file before encoding contributes to higher quality sound. For example, a closely miked interview in a studio will encode better than an interview conducted on a busy or windy street. For an in-depth discussion concerning pre-processing audio, see "Getting the Best Sound Out of RealAudio" on the Progressive Networks Web site at:

`http://www.realaudio.com/help/content/audiohints.html`

There are a number of shareware software packages available for recording sound, such as Cool Edit for Windows, available at

`http://www.netzone.com/syntrillium`

or you can use on of the many on-line search engines to find sound editing software for other platforms.

The audio file used for input to the RealAudio Encoder should use the full range of available amplitude, without exceeding the maximum input level. "Clipping" results if this level is exceeded, and it is heard as a pop or click in the resulting file. When encoded, a RealAudio file with clipping will have high-frequency background noise or static.

If the full amplitude range is not used, the RealAudio files will sound rough. Use your audio editor's "Increase Amplitude" or "Increase Volume" command to adjust the range before encoding the file. Some sound editors have a "Normalize" function that will maximize levels automatically.

Keep in mind that audio files take up a lot of disk space before they are encoded. Once the files are encoded into the RealAudio format, they will use 60 kB/minute (3.6 MB/hour) of disk space.

[1]Please respect copyrights. The following is an excerpt from the RealAudio Encoder License Agreement: Recipient represents and warrants that Recipient shall encode only those materials for which Recipient owns the copyrights and/or other intellectual property rights, or for which Recipient has authority from each such owner of each such right. Recipient hereby agrees to defend, indemnify, and hold PN, its directors, officers, employees and agents, harmless from any and all claims, damages, and expenses (including attorneys fees and costs) of any nature arising out of or relating to any breach of foregoing warranty, including but not limited to claims based on invasion of privacy, right of publicity, defamation, or other causes of action.

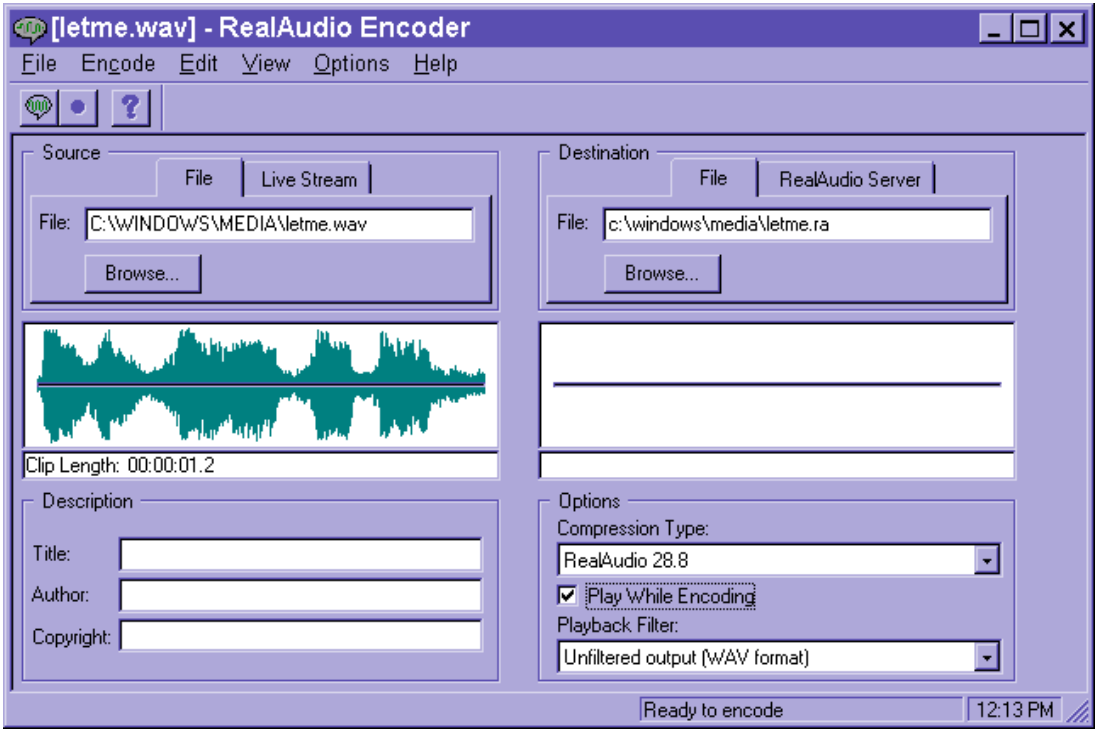
Creating RealAudio Files

The RealAudio Encoder allows you to encode files or live streams into RealAudio files or directly to the RealAudio Server for live broadcasts. The Encoder encodes files in real-time. You can browse folders and select files for encoding, or drag and drop files directly onto the Encoder icon. The Encoder window displays information about the audio file's sampling rate before and after compression. You can also enter author, title, and copyright information. This information will then appear in the RealAudio Player window when the user plays the file.

The RealAudio Encoder

To open the Encoder, double-click on **encoder.exe**. The Encoder window is comprised of the menu and toolbar, an encoding status display, and four information panels: Source, Destination, Description and Options. Figure 1-1 shows a window with information already partially entered.

Figure 1-1. RealAudio Encoder Window



Menu Bar

The Menu Bar consists of the following Menus and options:

File Menu

- Clear -- Clears the Encoder window.
- Exit

Encode Menu

- Start Encoding
- Stop Encoding

Edit Menu – these are standard text editing functions for the text fields in the Source, Destination and Description panels.

- Undo
- Cut
- Copy
- Paste

View Menu – selecting or clearing these options displays or hides the following parts of the Encoder:

- Toolbar
- Status Bar

Options Menu

- Show Audio Signal – Checking this option toggles between an audio signal display graph and an encoding progress bar.
- Save Default Settings – Checking this option saves the current settings in the Options and Description Panels.
- Network Preferences – Only available in the Live Encoder.
  - Outgoing buffer size – the size of the pre-network buffer.
  - Preferable Packet Size – the minimum packet size the Encoder will send to the buffer.

Source and Destination Panels

The Source panel allows you to choose the type of input source you will be using, either File or Live Stream.

- **File.** A previously digitized audio file; indicates where the input source is located.

- **Live Stream.** A live feed from an event or a studio, or audio from a CD, DAT tape, satellite feed or other high quality source; indicates the sampling rate and resolution of the live stream.

The Encoder automatically detects the audio type of the source file. If for some reason the Encoder can not recognize the file type, you will be presented with a dialog box that allows you to select the sampling rate, format and resolution of the source file.

The Destination panel has two tabs, File and RealAudio Server. If you do not have the Live Encoder, only the File tab will be enabled. The File tab

allows you to specify the location of the .ra output file. If you are encoding from a file, the Encoder suggests a default output file with the same name (with a .ra file name extension) in the same location as the source file. If this is not the desired location, or you are encoding from a live source, enter the appropriate destination path and the desired filename. If you are using the Live Encoder for a live broadcast, select the RealAudio Server Destination tab. Chapter 3 *Producing Live Events* provides more information on RealAudio Server tab settings.

Progress Information

The central area of the Encoder window displays the progress of the encoder. The information is displayed in both graphical and numerical form. You may choose to view the progress in two different ways, as a waveform or as a straight bar, by toggling Show Audio Signal under the Options Menu. If you select Show Audio Signal, the left graph shows the original audio signal while the right graph displays the decompressed waveform of the .ra file being created.

Description Panel

The lower left side of the Encoder window has a Description panel for the RealAudio file. You have the option of providing the title, author and copyright information about the audio file into the corresponding areas. This information will then be displayed in the RealAudio Player and the RealAudio Plug-in. For example, if you type in the following information into the Encoder window:

```
Title:      ABC News 3 PM 19 July 1995 Pacific Time
Author:     ABC News and Information Network
Copyright:  1995
```

When playing the clip, the Player will display the information as shown in Figure 1-2.

Figure 1-2. File information displayed by the RealAudio Player

You can change the description field on a file you have already encoded by re-encoding the file with the new description information. To do this select the .ra file to change from the Source Panel File tab, type in your new description and select the Start Encoding command from the Encode menu or click on the Start Encoding button on the left of the toolbar.

Options Panel

The lower right portion of the Encoder window allows you to set certain Options for encoding your files. This is where you select the Compression Type desired for compressing your audio file. There are two options available: 14.4 and 28.8. The 14.4 compression is best optimized for speech. It yields a file that plays with AM quality, and which can be played over a 14.4 Kbps modem. The 28.8 compression supports a greater dynamic range and more complex audio sources than 14.4 compression. 28.8 format files require at least a 28 Kbps modem connection to the Internet. If you choose 14.4 compression, you will need to make sure that you have 1.0 kB of disk space available per second of audio. The 28.8 compression requires 1.6 kB of disk space per second of audio.

If you would like to listen to the audio file as it is being encoded, you may do so by checking the “Play While Encoding” option in the Options panel. Using this feature, however, may slow down the encoding process. You can choose to listen to either the input file or the decompressed .ra output file.

Starting the Encoding Process

After you have selected your input file and output destination, set your compression type and entered the desired description information, you are ready to begin the encoding process. To encode, select Start Encoding from the Encode Menu or click on the Start Encoding button on the left of the toolbar.

Once you are familiar with the standard method of encoding you may find that drag and drop encoding or command-line encoding will work for your purposes.

Drag and Drop Encoding

RealAudio supports two types of drag and drop encoding. Clicking on a file and dragging it on to an open Encoder will place the path and filename information into the File field in the Source tab. Enter the description information and other desired settings and click the Start Encoding button to start the encoding process.

To initiate the encoding process, you can also click on the file that is to be encoded, and drag and drop it onto the **encoder.exe** file. The Encoder window will be open only during the encoding process, so you will not be able to set or change the encoding options. The file is encoded using the source file name and with the last saved default settings. This means that if you have selected Save Default Settings on the Options Menu, the saved settings will determine the compression type, the description information and playback filter used during encoding. If you have not selected Save Default Settings your file will be encoded in RealAudio 28.8 format.

Command-line Encoding

To automate the encoding of multiple audio files, you can specify command-line arguments that direct the Encoder to process an input file and then shut down. The syntax is:

```
ENCODER  <file to encode>  [<settings file>]
```

where `<file to encode>` is the digital audio file to encode and `<settings file>` is the name of an optional settings file. For example,

```
ENCODER  complete.wav  intro.txt
```

would encode the file `complete.wav` with the Encoder settings specified in the file `intro.txt`.

The `<settings file>` allows you to specify the compression type and the title, author and copyright strings for the `.ra` file. If not specified, the settings are those most recently stored by selecting Save Default Settings from the Options menu. The settings file is a text file with the following format:

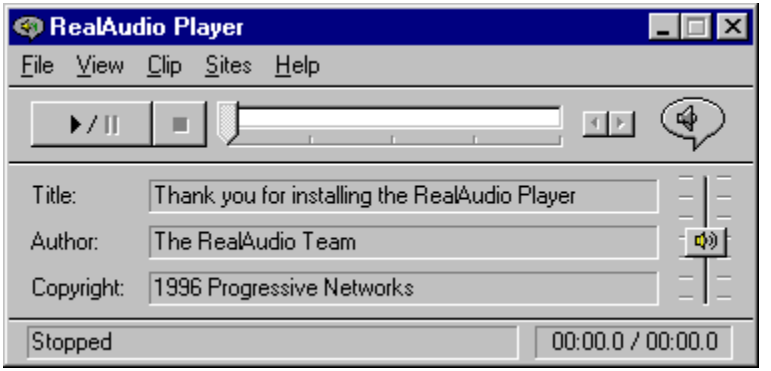
```
[Default Settings]

title = Title String Here

author = Author String Here

copyright = Copyright String Here

Compression = <1 for 14.4 compression or
               2 for 28.8 compression>
```

The settings file must start with the [Default Settings] line. For example, if the settings file `intro.txt` contains the following:

```
[Default Settings]

title = Thank you for installing the RealAudio
Player.

author = The RealAudio Team

copyright = 1996 Progressive Networks

Compression = 1
```

the RealAudio Player will display the information contained in `intro.txt` as in Figure 1-3.

Figure 1-3. Player display of `intro.txt`



If you place the files you wish to encode and their corresponding settings files in the same directory as the RealAudio Encoder, you will not have to specify the file path. However, if you would like to encode files in other directories, you will need to specify the complete paths to all input files. You will also need to have the RealAudio encoder in your path. For example,

```
c:\raencod\ENCODER c:\files\welcome.wav
   c:\settings\intro.txt
```

Cutting and Pasting RealAudio Files

The Encoder is distributed with two command-line utilities, **racut.exe** and **rapaste.exe**, which enable you to edit RealAudio files. **racut.exe** creates a copy of a specified portion of your original file.[2]. You can use the status bar on the Player to determine the time frame of the portion of the file you want to copy. The command uses the following format:

```
racut output.ra input.ra startTime endTime
```

where `output.ra` is the name of the file to which you want to save the cut portion, `input.ra` is the file from which to cut and `startTime` and `endTime` are the times between which you want to cut out. The time format is:

```
[[[days:]hours:]minutes:]seconds[.tenths]
```

[2] Cutting files encoded in the 28.8 format can only be done in blocks of 1.44 seconds; if your selected time is not a multiple of 1.44 the utility will round down to the nearest acceptable time frame.

where bracketed arguments are optional. Seconds are the only required value. The time will be interpreted from right to left, so it is not necessary to put a zero into every blank field, unless you are missing an increment between two other increments. For example, if you want to cut from the sixth hour of the first day (set of 24 hour periods) to the third day at four hours and one tenth of a second, your command would look as follows:

```
racut output.ra input.ra 06:00:00 02:4:00:00.1
```

rapaste.exe is a utility that creates a new .ra file from two or more .ra files. The command uses the following format:

```
Rapaste output.ra input1.ra input2.ra [… inputn.ra]
```

where `output.ra` is the name of the file you are creating, `input1.ra` is the first file you want in the new file, and `input2.ra` is the file to be appended to the end of `output.ra`. The number of input files is limited to the system command-line limits.

Splicing a file into the middle of an existing file is a multi-step process. First use **racut.exe** to create a file that contains the audio up to the point at which you want to insert a new file. Then use **racut.exe** to create a file with the audio to follow the inserted audio. Finally, use **rapaste.exe** to combine all three files together. For example, the syntax to put `insert.ra` into `original.ra` at 25 minutes would look as follows:

```
racut part1.ra original.ra 00.0 25:00

racut part2.ra original.ra 25:01 50:00

rapaste newfile.ra part1.ra insert.ra part2.ra
```

`newfile.ra` would then consist of `original.ra` with `insert.ra` inserted 25 minutes into `original.ra`.[3] To protect your original files, you should always work with a copy of your original.

Bandwidth Negotiation

Bandwidth negotiation is a Server feature that allows visitors to your Web site to automatically receive the highest quality format available to their

[3] Pasting files encoded in the 28.8 format can only be done in blocks of 1.44 seconds; if your selected time is not a multiple of 1.44 the utility will round down to the nearest acceptable time frame. This may result in gaps in the audio or the inclusion of unintended audio.

computer, while only requiring one generic link to a particular content file on your Web site. For example, without bandwidth negotiation, to provide content in both 14.4 and 28.8 formats, your Web site would require two Hypertext links and two metafiles, one to a 14.4 file and one to a 28.8 file. Using bandwidth negotiation, however, you would only need one Hypertext link and one metafile. For more information on bandwidth negotiation see the *RealAudio Server Administrator's Guide*.

Chapter 2: Web Site Configuration

This chapter covers metafiles and the Web site configuration necessary to provide RealAudio content. If you have created HTML documents in the past, you know that links to other Web pages are shown in the HTML document as the URL of the document you wish to access. Since RealAudio files are located with the RealAudio Server and can only be retrieved by the RealAudio Player, you must create a link between the Web server and the RealAudio Server. This link is a RealAudio metafile.

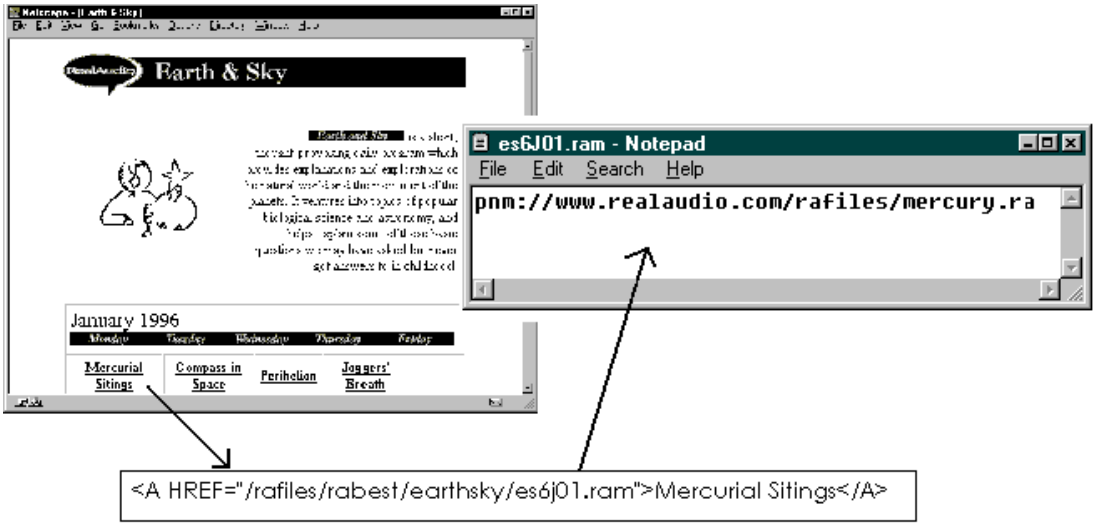
RealAudio Metafiles

Directly referencing a RealAudio file in a Web page would initiate a download of the file prior to playback. To play real-time audio, RealAudio files must be served by a RealAudio Server and played through a RealAudio Player. Metafiles are text files which contain the URL of a RealAudio file. They act as the link between the Web Server and the RealAudio Server.

Imagine your Web page as a sheet of paper containing directions to someone's house. In most situations, you would follow the directions and arrive at that house. For RealAudio files, a Web page is like a sheet of paper that contains directions to a house, and once you arrive at the house, there is another piece of paper attached to that door with a set of directions to another house, which is your actual destination.

When accessing RealAudio from a Web page, the first set of directions is represented by a link to a RealAudio metafile, recognized by a .ram or .rpm file name extension. The second set of directions represents a link from the metafile to the RealAudio file, recognized by a .ra file name extension. Instead of pointing to the audio file, your Web page will point to a metafile. Unlike the usual HTML link, the metafile does not display information through the browser, but rather, it provides the audio file's URL to the RealAudio Player. The Player uses this information to locate the .ra file on the RealAudio Server.

Figure 2-1. HTML link to a metafile.



Creating Metafiles

The metafile that you create will contain the address of the RealAudio (.ra) file or files being accessed by that link. The RealAudio System uses two metafile types: .ram and .rpm. Both types of metafiles are created in the same manner. The different file extensions, however, tell the Web browser with which application to play the accessed file:

.ram file – launches the RealAudio Player

.rpm file – loads a RealAudio Plug-in instance

Most URLs begin with a locator type that identifies the protocol being used to exchange information between the client and the server. Common locator types include http and ftp. The locator type used by RealAudio software is pnm. You must include pnm as part of the URL referenced in the metafile.

To create a metafile, open a text editor such as Notepad or Microsoft Word, and type in the URL of the RealAudio file using the pnm locator type and the following syntax:

```
pnm://hostname/path
```

For example, if you create a metafile to reference a RealAudio file called `hello1.ra`, the text of the metafile might read:

```
pnm://www.server.com/hello1.ra
```

If you want more than one file to play when the user clicks on a link, you can create a metafile that contains several URLs on separate lines (with no blank lines). For example,

```
pnm://www.server.com/hello1.ra
pnm://www.server.com/welcome1.ra
pnm://www.server.com/stayhere.ra
```

The Player will play these URLs in sequence. Listeners can use the Clip Menu on their Player to move forward and backward between clips in a multi-clip .ram file.

Save your file as a "Text Only with Line Breaks" type file using a .ram file name extension. For example, you may wish to save the metafile that references the above RealAudio files as

```
3track.ram
```

To insert a reference to this .ram file in your HTML, reference the metafile within a hyperlink. For example:

```
<A HREF="http://www.realaudio.com/3track.ram">Hello!</A>
```

Make sure that your Web server is properly configured to understand that the extension .ram refers to the MIME type **x-pn-realaudio** (detailed instructions for configuring a variety of Web servers are available in the *RealAudio Server Administrator's Guide*).

Customizing Links to Audio Content

Optional arguments are available to give you optimal control over what will be heard and seen on the RealAudio Player when someone clicks on your link. You may choose to start a RealAudio file at a point other than the beginning or to stop playing it before the actual end of the file. You may also change the Title, Author and Copyright information seen during playback.

These options follow the URL and are preceded by a ? and separated from each other by &. The syntax is as follows:

```
pnm://www.realaudio.com/test.ra?[option1]&[option2]
```

Changing Start and Stop Times

To create a link that starts playing the file from a point other than the beginning of the file append the desired start time to the URL. Use the following format:

```
start="dd:hh:mm:ss.ss"
```

Tenths of seconds are separated from seconds by a decimal point; and the other units of time are always separated by colons. The time will be interpreted from right to left, so it is not necessary to put a zero into every blank field, unless you are missing a step between two other unit. For example,

```
pnm://www.realaudio.com/test.ra?start="30"
```

would start playing `test.ra` from thirty seconds into the file.

To create a link that will play a file up to a specified end point other than the actual end of the file, append the desired end time to the URL, using the following format:

```
end="dd:hh:mm:ss.ss"
```

The specified amount of time is from the absolute beginning of the file. For example,

```
pnm://www.realaudio.com/test.ra?end="5:30"
```

would stop playing `test.ra` five minutes and thirty seconds from the beginning of the file.

To play only a designated section of a file append both the start time and the end time to the URL. For example,

```
pnm://www.realaudio.com/test.ra?start="30"&end="5:30"
```

would start playing `test.ra` from the thirty second point and stop playing after reaching the point five minutes and thirty seconds from the beginning of the file.

Changing the Title, Author and Copyright Information

Change the Description information that the Player and Plug-in will display by appending the desired change to the end of the URL. The formats are as follows:

```
title="title changes"

author="author changes"

copyright="copyright changes"
```

The Title, Author, and Copyright strings can be changed together or independently.

Changing the Description information in this manner does not change the Description information permanently. It is especially useful, for example, if you have one large .ra file that contains your band's entire CD but you want to play each song individually and credit the writer of each song. You could create a multi-clip .ram file as follows:

```
pnm://www.server/band.ra?end="5:30"&title="Song1"
pnm://www.server/band.ra?start="5:31"&end="7:45"&tit
le="song2"&author="Joe Smith"
pnm://www.server/band.ra?start="7:46"&end="15:01"&ti
tle="song3"&author="Jane Smith"&copyright="My Music,
1996"
```

RealAudio Plug-ins

The RealAudio Plug-in enables seamless integration of RealAudio into your Web page layout. You can place individual interactive components such as play buttons and volume controls anywhere on the page with the same level of control the HTML IMG tag offers for graphics. For an on-line demonstration of Plug-ins and associated HTML commands see the Progressive Networks Web page at:

`http://www.realaudio.com/products/ra2.0/plug_ins/index.html`

You can combine the RealAudio Plug-in with synchronized multimedia and frames to make your Web page more dynamic.

System Requirements

Viewing pages that use Plug-ins requires a Web browser that supports Netscape's Plug-in architecture. If you choose RealAudio Player 2.0 Express Setup the Plug-in will install automatically if there is a Plug-in compatible Web browser detected. During RealAudio Player 2.0 Custom Setup, you have the option of installing the Plug-in. Upon successful installation the file `npra16.dll` or `npra32.dll` will be located in your Web browser's plug-in directory.

Using the Plug-in: The Basics

The EMBED tag specifies Plug-in attributes in HTML pages in much the same way that the IMG tag specifies image attributes. The basic EMBED tag for RealAudio contains only the attributes SRC, WIDTH and HEIGHT, as shown below:

<EMBED SRC=metafile.rpm WIDTH=width_value HEIGHT=height_value>

For example:

<EMBED SRC="sample1.rpm" WIDTH=300 HEIGHT=134>

Do not place the EMBED tag within a table. For details on other available attributes, see *Custom RealAudio EMBED Tag Attributes* below.

The SRC Attribute

The SRC attribute specifies the RealAudio metafile to be accessed. As mentioned earlier, the RealAudio Plug-in has the .rpm file name extension. This file name extension tells the user's Web browser to load the RealAudio Plug-in rather than the stand alone RealAudio Player.

For the end user's Web browser to correctly identify the `.rpm` files, you or your system administrator must first configure the `.rpm` MIME type in your Web server. End users do not need to configure their Web browsers to recognize the .rpm MIME type. The plug-in architecture will automatically send `.rpm` files to the RealAudio Plug-in. Files with a `.rpm` extension are identical to `.ram` files except for the extension.

The WIDTH and HEIGHT Attributes

The WIDTH and HEIGHT attributes specify the size of the embedded RealAudio component. Unlike images, Plug-ins do not size automatically. The WIDTH and HEIGHT can be specified in pixels (the default) or as a percentage of the Web browser window (e.g. WIDTH=100%). If the WIDTH and HEIGHT attributes are not included, the Plug-in may appear in an icon-sized square space.

If you want your Plug-in component to maintain an absolute size, specify the size in pixels. If you want the Plug-in to scale with the Web browser window, specify the size as a percentage. For example, if you want to stretch the Position Slider and Status Bar to fit the entire width of the Web browser window, use a WIDTH=100% tag.

Using the <NOEMBED> Tag

Some Web browsers do not support Plug-ins. You can create HTML pages that are enhanced for Plug-ins and also work for other browsers by using the <NOEMBED> tag to include HTML statements that will appear in Web browsers that do not support Plug-ins.

The <NOEMBED> command should appear after an <EMBED> command and should have the following syntax:

<NOEMBED> HTML to be ignored </NOEMBED>

For example,

```
<EMBED SRC="sample1.rpm" WIDTH=300 HEIGHT=134>

<NOEMBED> <A SRC="sample1.ram"> Play the audio
using the stand alone Player! </A></NOEMBED>
```

would use the Plug-in if the your page is accessed by a browser supporting Plug-ins and display the message “Play the audio using the stand alone Player!” if the browser does not support Plug-ins.

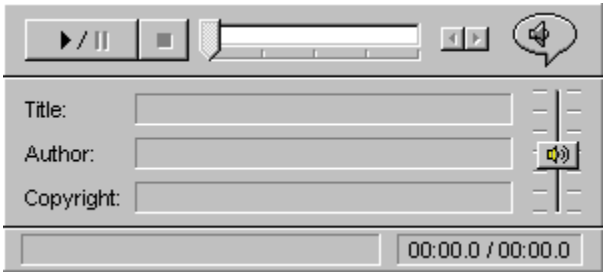
Custom RealAudio EMBED Tag Attributes

The attributes described below are specific to the RealAudio Plug-in.

The CONTROLS Attribute

You can use multiple EMBED statements to construct a custom interface made up of individual controls. The following values allow you to embed individual controls from the RealAudio Player directly into your HTML layout.

`CONTROLS = All` (Default view if no control is specified)

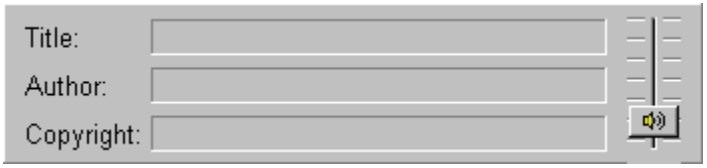


Embeds a full Player view including the Control Panel, Information and Volume Panel and Status Bar. This is also the default view if no CONTROLS attribute is specified.

`CONTROLS = ControlPanel`

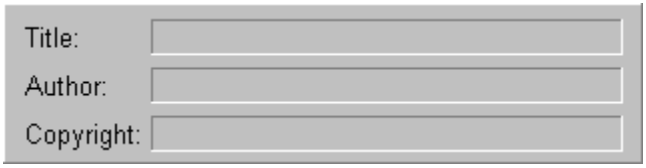
Embeds the Play/Pause button, the Stop button and the Position slider. This is similar to the panel the Player application displays when none of the options on the View menu are checked.

`CONTROLS = InfoVolumePanel`



Embeds the area Title, Author and Copyright information and a Volume Slider on the right-hand side. This is the same panel that the Player application displays when the Info & Volume option on the View menu is checked.

```
CONTROLS = InfoPanel
```



Embeds the area showing Title, Author and Copyright information only.

```
CONTROLS = StatusBar
```

Embeds the Status Bar showing informational messages, current time position and clip length. This is the same panel that the Player application displays when the Status Bar option on the View menu is checked. If you do not embed a Status Bar in your page, error messages will be displayed in the Web browser's status bar.

```
CONTROLS = PlayButton
```

Embeds the Play/Pause button only.

```
CONTROLS = StopButton
```

Embeds the Stop button only.

```
CONTROLS = VolumeSlider
```

Embeds the Volume Slider only.

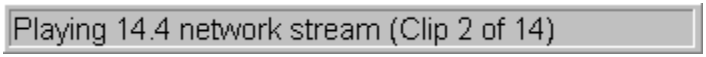
```
CONTROLS = PositionSlider
```

Embeds the Position Slider (scroll bar) only.

```
CONTROLS = PositionField
```

Embeds the field of the Status Bar showing Position and Length.

```
CONTROLS = StatusField
```



Embeds the field of the Status Bar that displays message text.

The NOLABELS Attribute

If your Web page is localized for a language other than English, or you use wish to the description fields to display information other than title, author, and copyright, you may want to remove the Title, Author, and Copyright labels from the information area. Controls that display the Title, Author and Copyright information for a clip, support a NOLABELS=TRUE attribute.

For example, using the default

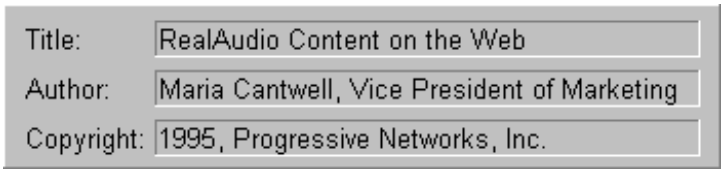
```
<EMBED SRC="use_lbl.rpm" WIDTH=350 HEIGHT=80
CONTROLS=InfoPanel CONSOLE=”Tester">
```

or NOLABELS=FALSE

```
<EMBED SRC="use_lbl.rpm" WIDTH=350 HEIGHT=80
CONTROLS=InfoPanel NOLABELS=false CONSOLE=”Tester">
```

will display the InfoPanel in Figure 2-2.

Figure 2-2. NOLABELS=FALSE (Default).

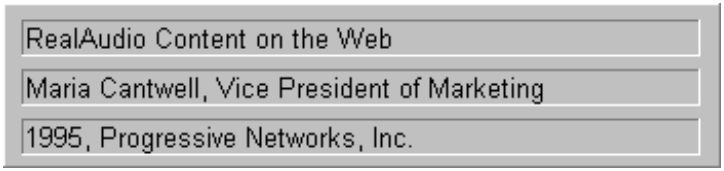


For example using NOLABELS=TRUE ,

<EMBED SRC="no_lbl.rpm" WIDTH=350 HEIGHT=80
CONTROLS=InfoPanel NOLABELS=true CONSOLE=”Tester">

produces the Info Panel displayed in Figure 2-3.

Figure 2-3. NOLABELS=TRUE.



The AUTOSTART Attribute

Adding an AUTOSTART=TRUE attribute to an EMBED tag tells the RealAudio Plug-in to automatically begin playing when the page is visited. You can use this feature to begin a narration or to play background music.

Since only one clip can play at a time, if you specify AUTOSTART for more than one Plug-in instance, only the last Plug-in to load will continue playing. The order in which the source files arrive is dependent on the Web Server and on your browser’s cache. This is not necessarily the order in which they are listed. You should use this attribute for only one Plug-in instance per page.

Making Plug-ins Work Together: the CONSOLE Attribute

The RealAudio Plug-in allows you to embed any number of RealAudio clips within a Web page. Normally each Plug-in behaves independently of all the others. Sometimes, however, you may want to link two or more Plug-ins together. For example, you might have a Play Button and a Position Slider that work as a pair. To include multiple components that work together,

specify a CONSOLE attribute and assign the same value to each instance. For example:

```
<EMBED SRC="sample1.rpm" WIDTH=30 HEIGHT=33
CONTROLS="PlayButton" CONSOLE="Clip1">

<EMBED SRC="empty1.rpm" WIDTH=300 HEIGHT=33
CONTROLS="PositionSlider" CONSOLE="Clip1">
```

Specifying a `CONSOLE` value of `"_master"` links the Plug-in to all other RealAudio Plug-ins on the page. Use this value when you want a control such as a Status Bar to display information for all audio clips. For example:

```
<EMBED SRC="sample1.rpm" WIDTH=300 HEIGHT=33
CONTROLS="StatusBar" CONSOLE="_master">
```

Because many platforms, including Windows, only support one volume setting for all digital audio, all volume sliders act on all clips and are automatically kept in synch regardless of the CONSOLE name. For this reason, you may wish to include only one Volume Slider on your page, with no reference to a `.ra` file in the .rpm file. You can create a dummy `.rpm` file (called something like `empty1.rpm`) to represent the SRC for the Volume Slider. Since a completely empty `.rpm` file may cause a browser to crash, the dummy `.rpm` file must contain a hard return (ASCII 13).

Each Plug-in instance (each EMBED tag) must have a unique SRC tag. If two EMBED tags have the same source, only the first will appear, and the other will be ignored in the HTML.

Using Plug-ins with HREF Tags

You can also use a `.rpm` file as the HREF for a normal Hypertext anchor. The Web browser will display the complete Player view at the full size of the browser window. The format is:

<A HREF="sample1.rpm">Play sample clip full-screen!</A>

In general, it is better to use an embedded control or a `.ram` file that launches the RealAudio helper application.

Making the Most of Your Content

To make the most of your RealAudio content, it is important to inform visitors to your page that they can listen to your site in RealAudio.

Let people know that they can listen to your audio instantly, without download delays; identify each audio clip with a RealAudio bubble icon (Figure 2-4) to distinguish it as real-time audio.

Figure 2-4. RealAudio bubble icon.

Include information on the software (the RealAudio Player) and the hardware (a sound card) that visitors will need to hear your audio. The easiest way to ensure that people listen to your audio is to make it easy for them to get the RealAudio Player. The RealAudio Web page *How to Design a RealAudio Site* located at

`http://www.realaudio.com/help/content/designwebsite.html`

contains a color icon (Figure 2-5 shows a black and white version) you can copy and use on your Web site. Link this icon to RealAudio's 2.0 download page.

Figure 2-5. Link to RealAudio download page.

Chapter 3: Producing Live Events

The Standard RealAudio encoder can create .ra files from live sources. The Live Encoder provided with the RealAudio Server has an additional capacity. In conjunction with the use of the Live Transfer Agent (LTA), it enables you to broadcast a live event directly to the Internet. There is also a utility, Simulated Live Transfer Agent (SLTA), that can take prerecorded material and broadcast it as if it were live. This chapter covers the use of the Live Transfer Agent and the Simulated Live Transfer Agent.

Server Setup for Live Events

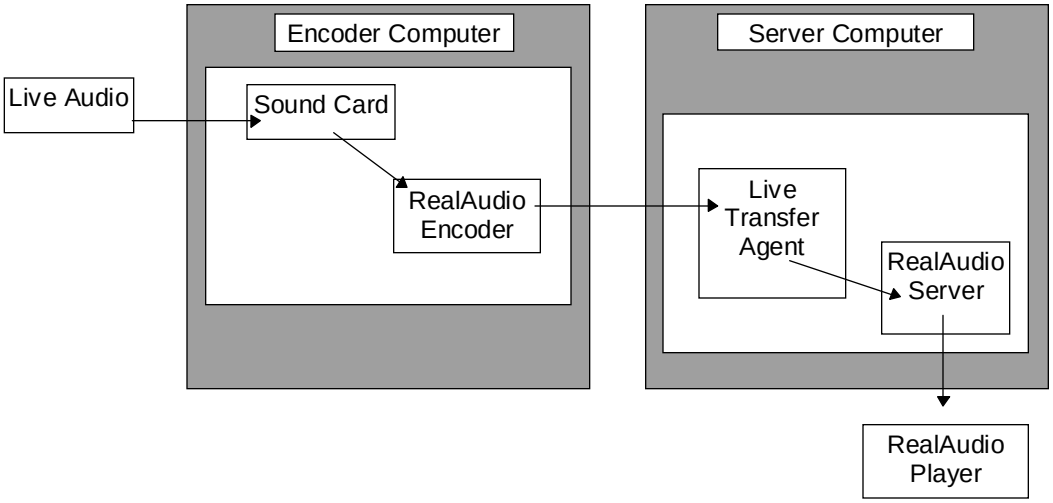
Live events can be sent over the Internet in real-time using the new RealAudio Live Encoder and the Live Transfer Agent (LTA) Application included with the RealAudio Server. To set up the RealAudio System for live events, the machine that has the RealAudio Encoder needs to be located where the sound is to be transmitted from, and the machine that runs the LTA needs to be located where there is enough bandwidth available for transmission to the Internet. There needs to be a TCP/IP network connection between these two machines.

When live events are to be sent directly from the Encoder to the RealAudio Server for delivery over the Internet, they need to be processed through the LTA before being sent to the RealAudio Server. The Live Transfer Agent must be located on the same machine as the RealAudio Server.

Live audio is fed into the Encoder's computer through a line that runs from a microphone to the microphone input jack or through a line that runs from a mixing console, CD player or other line level output device to the line input jack on the computer's audio card (see Figure 3-1). To ensure high quality .ra files, verify that you are using the proper input for the source you are using and that your input volumes are adjusted properly.

As soon as the audio is encoded, it is sent directly to the Server computer, where it is processed by the LTA and delivered to the RealAudio Server. The Server sends RealAudio packets to users who are listening to the live event through a link on a Web page.

Figure 3-1. Live Events Data Flow.



The LTA is a command-line application with the following usage:

```
lta -b <path/to/rafiles> -p <port#> -w <password>
```

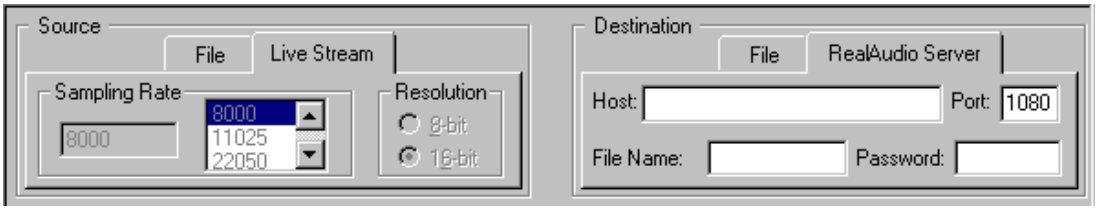
where `<path/to/rafiles>` is the path to the directory where the LTA will place the encoded files received from the Encoder, `<port#>` is the TCP port on which the RealAudio Server receives files from the Encoder, and `<password>` is the password shared by the LTA and Encoder. The password protects the live event from being accessed by outsiders during broadcasting. If the `-p` command is left out of the command-line, the default port 7069 will be used. To see options available for the LTA command-line (see also Appendix A, *Man Pages*), type:

```
lta -v
```

Once the LTA is started, the RealAudio Encoder can be opened and set up to start the encoding process. The encoding process occurs in real-time. As soon as audio is fed into the Encoder, it is encoded and sent to the Server computer.

In the Encoder, select the Live Stream tab from the Source panel, and the RealAudio Server from the Destination panel (see Figure 3-2).

Figure 3-2. Source and Destination Panels: Live Event Delivery Setup.



Currently the only settings available for Live Stream are 8000 Hz Sampling Rate and 16-bit Resolution.

If you would like to have a description of the live event displayed by the RealAudio Players that connect to it on the Web site, then you will need to fill in the Description panel in the lower left hand side of the Encoder.

The Destination panel of the Encoder must contain the same information that you used when starting the LTA. If any of the information in this panel does not correspond with the information you specified when starting the LTA, the encoded data will not be received by the LTA. The **Host** is the name of the computer where the Server resides. The **File Name** will be used in a .ram file to access the live event. It will also be the name of the RealAudio file once the live event finishes. **Port** is the port number that the Encoder will use to send the encoded .ra data. **Password** must be the same password used to start the LTA.

Options for encoding a file are found on the lower right side of the Encoder. You must choose the Compression Type that is needed for the live event. 14.4 compression will allow people with 14.4 Kbps or faster modems to listen to the event, while 28.8 compression limits the audience to those who have 28.8 Kbps or faster modems. Note that the 14.4 algorithm was optimized for speech transmission and is not recommended for music applications. Since the 28.8 algorithm can deliver more data it will provide higher quality audio. You will need 1.0 kB of disk space per second of audio when encoding using 14.4, and 1.6 kB per second of audio when encoding using 28.8.

To start the live broadcast, select the Start Encoding command from the Encode menu or click on the Start Encoding button on the toolbar.

To end the live broadcast, select the Stop Encoding command from the Encode menu or click the Stop Encoding button on the toolbar. When the LTA stops receiving data, it sends important header information to the RealAudio file which enables the file to be accessed later. If a break in the

transmission occurs and this header is not written, the file will not be a usable RealAudio file.

If you restart the LTA with the same path and file name as a previous event, the LTA will overwrite the existing `.ra` file. This is an important consideration. Reusing the same output file name can simplify Web page maintenance since the `.ram` file for a recurring event would not need to be changed. For example, if you broadcast news at 6:00 every night, you could have the LTA write to a generically named file `6pmnews.ra`. You could reference `6pmnews.ra` in your `.ram` file, and your Web page could always be linked to the same, unchanged `.ram` file. If, however, you wanted to maintain an archive of your broadcasts you would need to rename the `.ra` on a daily basis since the LTA will overwrite the existing `.ra` file.

Live Events with RealAudio 14.4 and 28.8

In order to reach the largest audience you will probably want to encode both a 14.4 and a 28.8 version of the event you are broadcasting. To accomplish this you will need to run two simultaneous Encoders and LTAs as well as provide two links on your Web page. The Encoders should be set up with different port numbers and output file names.

Prerecorded “Live” Delivery

The RealAudio 2.00 Server allows for the “live” delivery of non-live events through the SLTA (Simulated Live Transfer Agent). It takes a previously encoded .ra file and provides a live-like stream to the Server for distribution to RealAudio Players.

A RealAudio Player will see clips broadcast using the SLTA as a "live" clip and will always connect at the current point of the clip, which is offset from the time when the SLTA was started.

The SLTA is run from the command-line using the following arguments:

```
   bin/slta -b <basepath> -i <nameofinputfile> -o
<nameofoutputfile>
```

The <basepath> should be the same basepath the Server is using. `<nameofinputfile>` is the previously encoded `.ra` file you want to

broadcast. If -o <name of output file> is omitted, the name will default to `live.ra`

The RealAudio Server waits 30 seconds after the end of the file to give the Players time to disconnect and then it deletes the output file.

Chapter 4: Synchronized Multimedia

With the RealAudio System, you can provide real-time audio on-demand. In addition to basic audio content, RealAudio allows you to create real-time on-demand multimedia presentations using the `cevents` utility (provided with the RealAudio Server). These presentations can be as simple as a narrated slide show of your home page or as intricate as a multi frame training program that the viewer controls. This chapter covers the basics of creating single page and multi-frame events.

Creating Synchronized Multimedia Events with RealAudio

One of RealAudio's features is the ability to synchronize Web pages with audio. Thus the audio can be used as a "time line" to display new pages or frames in the Web browser or to update its content, enabling the creation of Internet slide shows, presentations, guided tours and site walk-throughs. The end user can have full random access (fast forward and rewind), and the Web browser content will automatically be synchronized with the audio.

The RealAudio system stores the information for the synchronized events in a file with a `.rae` file name extension. This file is automatically located by the RealAudio Server when the listener opens the `.ra` file. The RealAudio Server streams audio and event information to the Player. As the event information is streamed to the RealAudio Player, the RealAudio Player then sends information to the Web browser telling it when to update the page's content.

The RealAudio Plug-in will work with synchronized multimedia events. However, since sending the Web browser to a new URL will unload the RealAudio Plug-in when the HTML page is unloaded, it is best to create separate frames for the RealAudio controls and for the changing images (See *Frames: The Basics* below).

Currently, RealAudio does not support synchronized multimedia events with live audio.

Creating a synchronized multimedia event is a three-step process:

1. Use a text editor to create an input file specifying the display time for each URL.

2. Use the `cevents` command-line tool to generate a binary file from the text input file.

3. Place the resulting `.rae` file in the same directory as the `.ra` audio file.

Creating the Input File

Begin by creating a list of the URLs that you want to be shown during your presentation and the times within the RealAudio clip when they should be displayed. The syntax for each entry should follow the format:

```
u<space>starttime<space>endtime<space>URL for
event
```

Where `u` stands for URL event (each line starts with the letter `u`), `starttime` is the time into the audio clip when the new event is shown, `endtime` is the time into the audio clip when that event ends, and `URL` (generally beginning with "http:" or "file:") is the URL for that event (usually an HTML document). The time format is:

```
[[[days:]hours:]minutes:]seconds[.tenths]
```

The lines of the input file must be in ascending order of start time. The end time should be at least one tenth of a second before the start time of the next event. The following example shows how an input file might look:

```
u 00:00:10.0 00:00:59.9 http://www.RealAudio.com/
u 00:01:00.0 00:02:00.0 http://www.mysite.com/
page2/
```

This input file tells the Player to send the Web browser to the RealAudio home page at ten seconds into the audio clip. At one minute into the audio clip, the Web browser will display a page from "www.my_site.com".

The input file may also contain comment lines beginning with the # symbol. These comment lines are ignored by the event creation tool and are a good way to document the date that the file was created.

Generating an Event File Using the CEVENTS Command-Line Tool

You will need to create a RealAudio binary event file from the text file you just created. The command-line utility `cevents` is provided for this purpose. It uses the following syntax:

```
cevents <input text file>  <output event file>
```

where `<output event file>` is the same name as the audio file with which it will be associated. For example, to a create synchronized multimedia presentation to accompany `paradise.ra`, you would generate `paradise.rae` using the following command:

```
cevents paradise.txt paradise.rae
```

Placing Event Files on Your Server

Copy the `.rae` file to the directory as your `.ra` file. The `.ra` and `.rae` files must have the same name except for the file name extension. The RealAudio Server will automatically detect the file and send the event information to the Player, which then sends it the Web browser.

Using Synchronized Multimedia from Local Files

The RealAudio Player can also read local `.rae` files just as the Server does. To test the `.rae` file locally, place it in the same directory in which the corresponding `.ra` file is found. The URLs within your input file for CEVENTS can refer to local files. When passed to the Web Browser, partial path names will be interpreted as relative to the current working directory. For maximum reliability, specify complete paths as shown below:

```
u 10.0 45.0 file:c|/media/test2.html
```

Frames: The Basics

Some Web browsers support a feature which allows the browser window to be divided into multiple regions. Each region is called a *frame* and can

display a separate URL such as a graphic or document. For an introduction to frames visit Netscape's Web site. If you are unfamiliar with frames the following two sites provide a good introduction:

Understanding Frames:
`http://www.netscape.com/navigate/understanding_frames.html`

Frames: An Introduction:
`http://www.netscape.com/assist/net_sites/frames. html`

Frame Document

A frame document describes the sub-HTML documents or frames that will make up a window. The basic structure of a frame document is similar to that of a normal HTML document except that the FRAMESET tag replaces the BODY tag. Each frame is defined by the FRAME tag. In order to effectively use RealAudio each FRAME tag needs the SRC and NAME attributes. The SRC points to the URL to be displayed in the frame. The NAME attribute assigns a name to the frame so that it can be targeted by links in other documents. The example below show a simple frame document that would create two frames:

```
<HTML>

<FRAMESET ROWS="105,*">

    <FRAME SRC="banner.html" NAME="banner">

    <FRAME SRC="lyrics.html" NAME="Lyric">

</FRAMESET>

</HTML>
```

Synchronizing frames and audio

Frames and RealAudio content are synchronized in the same way that a regular Web page and RealAudio content are synchronized: a `.rae` file. The difference lies in the addition of the targeted frame name to the text file that is used to create the `.rae` file. The syntax for each entry should follow the format:

```
u<space>starttime<space>endtime<space>&&framename&&<spa
ce>URL for event
```

Where `u` stands for URL event (each line starts with the letter `u`), `starttime` is the time into the audio clip when the new event is shown, `endtime` is the time into the audio clip when that event ends, `&&` is a delimiter, `framename` is the frame name as specified in your frame document, and `URL` (generally beginning with "http:" or "file:") is the URL for that event (usually an HTML document). The time format is:

```
[[[days:]hours:]minutes:]seconds[.tenths]
```

The lines of the input file must be in ascending order of start time. The end time should be at least one tenth of a second before the start time for the next event.

The example below shows an `.rae` input file for a Web site that displays lyrics as a song plays. Each verse is displayed within a frame called LyricFrame.

```
u 00:00:10.0 00:00:35.0
&&Lyric&&http://www.songs.com/ver1.html
u 00:00:35.0 00:00:50.0
&&Lyric&&http://www.songs.com/ver2.html
```

More Information on frames

A complete discussion of frames is beyond the scope of this manual. For more information on frame attributes and syntax and on targeting windows consult the following Netscape Web pages:

Frames-- Syntax:

`http://www.netscape.com/assist/net_sites/frame_syntax.html`

Targeting Windows:

`http://www.netscape.com/eng/mozilla/2.0/relnotes/demo/target.html`

Appendix A: Man Pages

lta

Name

lta - RealAudio Live Transfer Agent

Synopsis

lta [-v] [-b basepath] [-p port] [-w password] [-s packetsize] [-m] [-q]

Description

The lta is used to broadcast a live event from the RealAudio Server. The RealAudio encoder connects to the LTA via TCP and the LTA creates a RealAudio file for the Server to serve.

The following options are available

-v prints out the version information of the lta. This includes the platform, build and release tags used to identify a particular release.

-b **basepath** The directory, **basepath** is where the lta will create the RealAudio file for the server. It is usually set to the same value as BasePath in the server configuration file.

-p **port** The lta will listen for incoming connections on port, **port**. This is the port that is used in the Live Encoder.

-w **password** The password, **password** supplied here is checked against the incoming password from the Live Encoder. If

they match the connection is accepted otherwise the connection is dropped.

-m Create Bandwidth Negotiation compatible directory and files for any requested file names. This means a directory is created containing the appropriate encoder and bandwidth named file.

-q Quit after encoding a single file. Otherwise, the lta will accept and create files for multiple sequential connections.

-s **packetsize packetsize** sets the size of the packet that is read from the network. This can be changed if connection problems are observed. The standard size for incoming packets is 240. This can be increased on a LAN connection if packet loss is occurring. This must also be changed in the Live Encoder.

slta

Name

slta - RealAudio Simulated Live Transfer Agent

Synopsis

slta [-v] [-b basepath] -i input [-o output]

Description

The RealAudio slta is used to turn a existing RealAudio file into a RealAudio file that can be server as a live event. This can be used to provide delayed broadcast of a live event.

-v	prints out the version information of the monitor. This includes the platform, build and release tags used to identify a particular release.
-b basepath	basepath specifies the directory where the output RealAudio file is created
-i input	the input file is the file that will be converted to a live RealAudio file.
-o output	this is the name give to the resulting file. The output file will be created in the base path directory.

cevents

Name

cevents - RealAudio Synchronized Multimedia Compiler

Synopsis

cevents inputfile outputfile

Description

cevents takes the supplied text file inputfile containing the Multimedia Event descriptions and converts it to a compiled multimedia file, outputfile.

File Format

The inputfile has the following format:

u <starttime> <endtime> <URL>

where each entry is on a single line with each value separated by spaces and where

<starttime> is the start time of the event in HH:MM:SS.t format
<endtime> is the end time of the event in HH:MM:SS.t format
<URL> is the URL of the page for the event

INDEX

A

amplitude
  adjusting, 4
  clipping, 4
audio files
  recording, 3
AUTOSTART, 25

B

bandwidth negotiation, 14
batch encoding. *See* command-line encoding

C

CEVENTS, 37
  input file, 36
command-line encoding, 10
compression, 3
CONSOLE, 25

D

drag and drop encoding, 10

E

editing .ra files, 12
EMBED, 20
  custom tag attributes, 22
encoder
  illustration, 6
  opening the encoder, 5
  using the encoder, 5
encoding
  command-line, 10
  drag and drop, 10
  live events, 29
  operation, 1
  pre-processing audio, 4
  starting, 10
extensions
  .rae, 35
  .ram, 15, 17
  .rpm, 15

F

frames, 38
  frame document, 38
  synchronizing with audio, 39

H

HEIGHT, 21
HREF, 26

L

live delivery
  of previously encoded files, 32
live events
  14.4 and 28.8 format, 32
  lta command, 30
  Server setup, 29
Live Transfer Agent, 30

M

metafiles, 15
  creating, 16
Multimedia. *See* Synchronized multimedia

N

NOEMBED, 21
NOLABELS, 24

P

plug-ins, 19
  system requirements, 20

R

racut.exe, 12
rapaste.exe, 13
RealAudio Player
  distribution address, 1
  links to obtain Player, 27
  operation, 1

S

Simulated Live Transfer Agent, 32
SLTA. *See* Simulated Live Transfer Agent
sound
  editing software, 4
Source Panel, 7
SRC, 20
Synchronized multimedia, 35
  CEVENTS, 37
  frames, 38
  from local files, 37
  placing on a Server, 37

W

WIDTH, 21